What is claimed is:

1   1. A method in a computer graphics system for rendering on an output device, having a finite

2   number of pixels, a non-zero thickness line segment with reduced aliasing, comprising:

3   expanding an edge of the line segment touching but not covering a pixel center of the line

4   segment to be rendered on the output device so that the expanded line segment covers the center

5   of the pixel touched;

6   determining, using the pixel centers, the pixels to be included in the expanded line

7   segment, the line segment being distinguishable from a background over which said line segment

8   is rendered by having a shade different from a shade of the background; and

9   for each pixel that is included in said non-zero thickness expanded line segment,

10   determining the area of the pixel partially or fully covered by said line segment;

11   and

12   based on the area of the pixel covered, determining a shading value for the pixel

13   by interpolating between the shade of said line segment and the shade of the background.

1   2. A method for rendering a non-zero thickness line segment as recited in claim 1,

2   wherein the output device has a x-y coordinate system established thereon and the pixels

3   of the output device each have centers with an x-y coordinate; and

4   wherein the step of expanding an edge includes moving the edge of said line segment by

5   an amount equal to (a+b)/2a in the x-direction to include the center of a pixel that has a corner

6   traversed by an edge of said line segment, wherein a is greater than or equal to zero and b is

7   greater than or equal to zero.

1   3. A method for rendering a non-zero thickness line as recited in claim 1,

2   wherein the output device has a x-y coordinate system established thereon and the pixels

3   of the output device each have centers with an x-y coordinate;

4   wherein an edge of said line segment has an equation ax + by + c = 0; and

5   wherein the step of expanding an edge includes altering the equation of an edge of said

6   line segment by adding an amount (|a|+|b|)/2 to the c parameter of the equation.

1   4. A method for rendering a non-zero thickness line as recited in claim 1,

2   wherein the output device has a x-y coordinate system established thereon and the pixels

3   of the output device each have centers with an x-y coordinate; and

4   wherein the step of determining the pixels to be included in the non-zero thickness line

5   includes:

6   evaluating the equation of an edge of said expanded line segment with the x and

7   y-coordinates of the center of each pixel; and

8   testing whether the result of the computation is greater than or equal to zero.

1   5. A method for rendering a non-zero thickness line as recited in claim 4, wherein evaluating the

2   equation of an edge of said expanded line segment with the x and y-coordinates of the center of

3   each pixel includes computing $ax_0 + by_0 + c + (|a|+|b|)/2$, where x_0 and y_0 are the coordinates of

4   the pixel center.

1   6. A method for rendering a non-zero thickness line as recited in claim 1,

2   wherein the shade of the pixel covered and the shade of the background are each

3   indicated by a numerical value; and

4   wherein the step of determining the shading value of the pixel by interpolating includes:

5   forming a first product of the shade numerical value of said line segment and a

6   fraction f representing the area of the pixel covered;

7   forming a second product of the shade numerical value of the background and a

8   fraction $(1-f)$ representing the area of the pixel not covered; and

9   adding the first and second products.

1   7. A method for rendering a non-zero thickness line as recited in claim 1,

2   wherein the output device has a x-y coordinate system established thereon and said line

3   segment has a slope factor sf related to the slope of said line segment and a parameter p

4   proportional to an x-distance between an edge of said line segment traversing a pixel and a pixel

5   boundary; and

6   wherein, for an edge of said line segment that traverses a partially covered pixel, the step

7   of determining the area of a partially covered pixel includes:

8   determining that the area covered is less than or equal to a first predetermined

9 limit; and

10   computing the triangular area covered by said line segment.

1 8. A method for rendering a non-zero thickness line as recited in claim 7,

2   wherein the parameter p is equal to the product of the slope factor sf and the distance

3 between an edge of said line segment traversing the pixel and a pixel boundary;

4   wherein the line segment has a slope m and the slope factor sf equals $m/(m+1)$; and

5   wherein the step of computing the triangular area covered by said line segment includes

6 forming a product $\frac{1}{2} * p^2 * (1\text{-}sf)^{-1} * sf^1$ to find the area.

1 9. A method for rendering a non-zero thickness line as recited in claim 1,

2   wherein the output device has an x-y coordinate system established thereon and said line

3 segment has a slope factor sf related to the slope of said line segment and a parameter p

4 proportional to an x-directed distance between an edge of said line segment traversing a pixel

5 and a pixel boundary; and

6   wherein, for an edge of said line segment that traverses a partially covered pixel, the step

7 of determining the area of a partially covered pixel includes:

8   determining that the area covered is greater than a first predetermined limit;

9   computing the maximum triangular area covered by said line segment;

10   computing the area of a parallelogram covered by said line segment; and

11   computing the sum of the maximum triangular area and the parallelogram area.

1 10. A method for rendering a non-zero thickness line as recited in claim 9, wherein the slope

2 factor sf equals $m/(m+1)$, where m is the slope of said line segment.

1 11. A method for rendering a non-zero thickness line as recited in claim 9, wherein the p

2 parameter equals the product of said x-directed distance and the slope factor sf.

1   12. A method for rendering a non-zero thickness line as recited in claim 11, wherein said x-
2   directed distance is computed as the quotient $ax_0+by_0 + c + (|a|+|b|)/2$ and a, where x_0 and y_0 are
3   the coordinates of the center of the pixel and the line segment edge has an equation $ax+by+c=0$.

1   13. A method for rendering a non-zero thickness line as recited in claim 9, wherein the first
2   predetermined limit is the maximum area triangular area covered by said line segment traversing
3   through the pixel.

1   14. A method for rendering a non-zero thickness line as recited in claim 9, wherein the step of
2   computing the maximum triangular area covered by said line segment includes forming a
3   product $\frac{1}{2} * (1-sf) * sf^1$ to find the maximum triangular area.

1   15. A method for rendering a non-zero thickness line as recited in claim 9, wherein the step of
2   computing the parallelogram area covered by said line segment includes forming a sum of $p*sf^1$
3   and $(1-sf^1)$ to find the parallelogram area.

1   16. A method for rendering a non-zero thickness line as recited in claim 1, wherein, for an edge
2   of said line segment that traverses a partially covered pixel, the step of determining the area of a
3   partially covered pixel includes:
4   determining that the area covered is greater than a second predetermined limit, leaving a
5   triangular area not covered;
6   computing the triangular area not covered by said line segment; and
7   computing the difference between one and the triangular area not covered to find the area
8   of the pixel covered.

1   17. A method for rendering a non-zero thickness line as recited in claim 16, wherein the second
2   predetermined limit is the sum of the maximum triangular area and the maximum parallelogram
3   area of said line segment traversing the pixel.

1   18. A method for rendering a non-zero thickness line as recited in claim 16, wherein the step of

2   computing the triangular area covered by said line segment includes forming a product $\frac{1}{2} * p^2 *$

3    $(1-sf)^{-1} * sf^{1}$ to find the triangular area covered.

1   19. A method for rendering a non-zero thickness line as recited in claim 1, wherein when two

2   parallel edges of said line segment traverse a partially covered pixel, the step of determining the

3   area of the partially covered pixel includes:

4   computing a first area of the pixel not covered by the first parallel edge;

5   computing a second area of the pixel not covered by the second edge; and

6   summing the first and second areas and subtracting the sum from one.

1   20. A method for rendering a non-zero thickness line as recited in claim 1, wherein when a first

2   edge along said line segment and a second edge orthogonal to said line segment traverse a

3   partially covered pixel, the step of determining the area of the partially covered pixel includes:

4   computing a first area of the pixel not covered by the first parallel edge and subtracting

5   the first area from one to form a first difference;

6   computing a second area of the pixel not covered by the second edge and subtracting the

7   second area from one to form a second difference; and

8   forming a product of the first and second differences.

1   21. A method for rendering a non-zero thickness line as recited in claim 1, wherein when two

2   parallel edges and a third orthogonal edge of said line segment traverse a partially covered pixel,

3   the step of determining the area of the partially covered pixel includes:

4   computing a first area of the pixel not covered by the first parallel edge;

5   computing a second area of the pixel not covered by the second parallel edge; and

6   summing the first and second areas and subtracting the sum from one to form a first

7   difference;

8   computing a third area of the pixel not covered by the third orthogonal edge and

9   subtracting the third area from one to form a second difference; and

10   forming a product of the first difference and the second difference.

75281.1.17

1   22. A method for rendering a non-zero thickness line as recited in claim 1, wherein, when two

2   parallel edges and a third and forth orthogonal edge of said line segment traverse a partially

3   covered pixel, the step of determining the area of the partially covered pixel includes:

4   computing a first area of the pixel not covered by the first parallel edge;

5   computing a second area of the pixel not covered by the second parallel edge; and

6   summing the first and second areas and subtracting the sum from one to form a first

7   difference;

8   computing a third area of the pixel not covered by the third orthogonal edge;

9   computing a fourth area of the pixel not covered by the fourth orthogonal edge;

10   summing the third and fourth areas and subtracting the sum from one to form a second

11   difference; and

12   forming a product of the first difference and the second difference.

1   23. A graphics pipeline for rendering a non-zero thickness line segment on an output device,

2   having a finite number of pixels, comprising:

3   an interpolator processor for computing a parameter proportional to a displacement

4   representing an x-directed distance between an edge of said line segment and the boundary of a

5   pixel that is partially covered by said line segment, the edge of said line segment having a known

6   edge relation that defines a slope factor for the edge, the pixel having a center with known x and

7   y coordinates; and

8   a shading processor for computing the area of the pixel covered by said line based on the

9   displacement parameter, the slope factor of said line segment, the edge relation and the x and y

10   coordinates of the pixel center.

1   24. A graphics pipeline as recited in claim 23, wherein said x-directed distance is computed as

2   the quotient $ax_0 + by_0 + c + (a+b)/2$ and a, where x_0 and y_0 are the coordinates of the center of

3   the pixel and the edge relation is $ax + by + c \geq 0$, and the slope factor is $a/(a+b)$.

1   25. A graphics pipeline as recited in claim 23, said displacement parameter is computed as sf *

2   $x_0 + (1-sf) * y_0 + sf * c/a + 1/2$, where x_0 and y_0 are the coordinates of the center of the pixel, the

3   edge relation is $ax + by + c \geq 0$, and the slope factor is $a/(a+b)$.

75281.1.17

1   26. A graphics pipeline as recited in claim 23,

2   wherein the pixel has center coordinates (x0, y0), the edge relation of the line segment is

3   $ax+by+c \geq 0$, and the slope factor is $a/(a+b)$;

4   wherein said interpolator process evaluates interpolator function, $Ps + Px * x + Py * y$, to

5   perform interpolation calculations; and

6   wherein displacement parameter is computed by setting Ps to $(sf * c/a + \frac{1}{2})$, Px to sf and

7   Py to $(1-sf)$ and evaluating the interpolator function at $(x0, y0)$.

1   27. A graphics pipeline as recited in claim 23, wherein the interpolator processor is a fixed

2   function single-instruction, multiple-data (SIMD) processor capable of operating on multiple

3   data items with the same instruction.

1   28. A graphics pipeline as recited in claim 23, wherein the shading processor is an instruction-

2   based computing element.